Express Mail No. EV 797 296 146 US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of: Chedgey, et al.	Confirmation No. 8068
Serial No.: 09/742,255	Art Unit: 2122
Filed: December 20, 2000	Examiner: Chuck O Kendall
For: SYSTEM AND METHOD FOR COMPUTER-AIDED GRAPH-BASED DEPENDENCY ANALYSIS	Attorney Docket No. 61134-0003

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Mail Stop AF
Commissioner for Patent
P. O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is to request pre-appeal brief review of the final Office Action of June 2, 2005 which rejected claims 20 and 21 under 35 U.S.C. 102 (b) as anticipated by Broekhuijsen (U.S. Patent No. 5,940,083), rejected claims 1, 11-19 under 35 U.S.C. 103(a) as unpatentable over Brotsky et al. (U.S. Patent No. 5,490,246) in view of Broekhuijsen, rejected claims 2 and 3 under 35 U.S.C. 103(a) as unpatentable over Brotsky et al. in view of Broekhuijsen and Blelloch (Provably Efficient Scheduling for Languages with Fine-Grained Parallelism, Published 1999), and rejected claims 4-9 under 35 U.S.C. 103(a) as unpatentable over Brotsky et al., in view of Broekhuijsen and Perttunen (U.S. Patent No. 6,359,635). Claims 1, 11, 13, 14 and 20 are independent claims.

Applicants' invention is directed to a Higraph structure for representing the relationships among elements of a complex system. A Higraph is a nested hierarchy of many-to-one mappings

from one directed graph to another directed graph with the same or fewer nodes and edges. (See pg. 14-16 of the specification.) At the bottom layer of the hierarchical Higraph structure is the "essential graph", which is a directed graph comprising nodes and directed edges. (See pg. 14, lns. 18-21.) Each node of the directed graph may represent, for example, a piece of software code in a software system, and each edge represents a dependency relationship between two pieces of software code. (See. e.g., Fig. 8.) The essential graph may be partitioned and each such partition may be mapped to a node in the next layer of the Higraph structure. A partition of these nodes in the next layer may be further mapped to a node in the next layer above. (See pg. 14, ln. 6-13, 18-22, a node in the next layer is referred to as a "meta-hinode" in pg. 14 ln. 22.) This "nesting" process may continue layer by layer until only one root node is left at the top layer of the hierarchy. During the nesting process, the edges between nodes in each layer are also mapped to edges between corresponding nodes in the upper layers. (See pg. 14, ln. 22-pg. 15, ln. 2.) Note that these mappings are used to recover the next lower layer when the user "unfolds" a nested layer.

An example of a Higraph structure is shown in Exhibit 1. A partition containing nodes F, G, H in the bottom layer is mapped to a node C in the next layer in yellow color. Note that nodes C, F, G, and H form a "vertical" subtree, in which node C is the "root node." These root nodes may be further partitioned and each such partition may be mapped to a root node in the next layer above. This "nesting" process may continue layer by layer until only one root node is left at the top layer. The edges between nodes in the bottom layer are also preserved. For example, there is an edge between root nodes C and D, because there exist edges 5 and 6 between the two partitions corresponding to nodes C and D in the layer below.

The Higraph structure and the preservation of relationships between root nodes of subtrees within the Higraph hierarchy are expressly recited in claims 20 and 21. Specifically, independent claim 20 requires that for each pair of subtrees not sharing any common node, there exists a relationship between the root nodes of said pair of subtrees, if there exists a relationship between a node in one subtree and a node in the other. Further, independent claim 20 requires that the relationship among the root nodes of the subtrees include a dependency relationship.

None of the prior art cited by the office action teaches, discloses or even suggests such a Higraph structure that preserves dependency relationships among subtrees. Broekhuijsen, the prior

art cited in the 102 rejection of claims 20 and 21, is directed to systems and methods for generating computer graphic images, such as curves. (See col. 1, lns. 5-10.) It teaches that an image with complex curves, or a "multi-curve object," can be created as a graph of nodes interconnected by edges or links. Each node represents a sequence of points of a curve to be selected by the user. Each node may also include reference pointers indicating the edges or links between nodes in the graph. (See, e.g. abstract and Fig. 4.) A user can then draw a complex curve by selecting the nodes along the edges.

Unlike applicants' invention, Broekhuijsen's graph is only two-dimensional. Further, there are no dependency relationships between the root nodes of different subtrees, because the root nodes of Broekhuijsen merely represent different curve segments for the user to chose from and the choice of one node does not depend on any other root node in the same graph. Furthermore, Broekhuijsen's graph is used to represent computer graphic images, not software entities as required by claim 21 of applicants' invention. It is clear from the specification that software entities include software code, such as C++ code, that can be executed by the computer. In contrast, graphic images, although generated by the computer, can never be executed by the computer. The examiner's argument that graphic images are software entities is inconsistent with the meaning of software entities in the specification. For these reasons, it is respectfully submitted that the rejection of claims 20 and dependent claim 21 as anticipated by Broekhuijsen should be withdrawn.

Likewise, it is respectfully requested that the §103 rejections be withdrawn, because none of the three references teaches, discloses or even suggests a Higraph structure in which software entities and their relationship are represented by a graph and the graph comprises a plurality of subtrees, each subtree representing one or more nodes in the graph, and the relationship among the subtrees represent edges among nodes in the graph, as required by independent claim 1. Neither does any of these references teach, disclose or even suggest a dependency analysis system comprising a Higraph structure, as claimed by independent claims 11, 13 and 14. Specifically, Brotsky teaches a graphical editor where the user creates a directed graph (the ACG) in order to describe how an image is constructed out of one or more figural elements such as graphical images. (See abstract and col. 5, lns. 4-20.) The ACG does not contain a tree structure, and the nodes of the ACG represent pieces of a graphical image, not software entities such as C++ code. Further, the edges between ACG's

nodes represent links between pieces of images, not dependency relationships between software entities such as C++ code.

Blelloch, another prior art cited by the office action, teaches a tool for dynamically scheduling the execution of tasks performed by parallel algorithms during program execution. (See, e.g., Abstract on pg. 281-282.) Blelloch does not teach a tree structure as claimed by applicants' invention. Further, the teachings of Blelloch serve a completely different purpose than Brotsky or Broekhuijsen, and thus can not be combined with Brotsky or Broekhuijsen in a 103 rejection. This is because Brotsky and Broekhuijsen, as described above, are directed to graphic editing tools. Blelloch, on the other hand, is directed to parallel computer program executors. It is obvious that a graph is not analogous to a task performed by parallel algorithms, neither is a graph executable by a computer.

Perttunen, yet another prior art cited by the office action, teaches the presentation of categorized information, such as organization charts, in the form of graphs and charts. See, e.g., col. 1, lns. 7-9. Like Brotsky, Broekhuijsen and Blelloch, Perttunen does not teach, disclose, or suggest a three-dimensional multi-layered tree structure that preserves dependency relationships among subtrees. It is also respectfully submitted that Brotsky, Broekhuijsen and Perttunen serve difference purposes. Thus, they are not analogous art and can not be combined in a 103 rejection.

Accordingly, it is respectfully requested that the 103 rejections of claims 1-9 and 11-19 be withdrawn. Further, in view of the forgoing remarks, the claims in this application are believed to be in condition for allowance. Such action is respectfully requested.

Respectfully submitted,

Date: December 2, 2005

Francis E. Morris (Reg. No. 24,615)
MORGAN, LEWIS & BOCKIUS LLP
Customer No. 048591
(212) 309-6000